



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/777,872	02/12/2004	William Preston Alexander III	AUS920030820US1	6694
35525	7590	04/10/2008	EXAMINER	
IBM CORP (YA)			VU, TUAN A	
C/O YEE & ASSOCIATES PC			ART UNIT	PAPER NUMBER
P.O. BOX 802333			2193	
DALLAS, TX 75380				
NOTIFICATION DATE		DELIVERY MODE		
04/10/2008		ELECTRONIC		

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ptonotifs@yeeiplaw.com

<b>Office Action Summary</b>	<b>Application No.</b> 10/777,872	<b>Applicant(s)</b> ALEXANDER ET AL.
	<b>Examiner</b> Tuan A. Vu	<b>Art Unit</b> 2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If no period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### **Status**

1) Responsive to communication(s) filed on 1/23/08.  
 2a) This action is FINAL.      2b) This action is non-final.  
 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### **Disposition of Claims**

4) Claim(s) 1-20 is/are pending in the application.  
 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.  
 5) Claim(s) \_\_\_\_\_ is/are allowed.  
 6) Claim(s) 1-20 is/are rejected.  
 7) Claim(s) \_\_\_\_\_ is/are objected to.  
 8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### **Application Papers**

9) The specification is objected to by the Examiner.  
 10) The drawing(s) filed on \_\_\_\_\_ is/are: a) accepted or b) objected to by the Examiner.  
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).  
 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### **Priority under 35 U.S.C. § 119**

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
 a) All    b) Some \* c) None of:  
 1. Certified copies of the priority documents have been received.  
 2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.  
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### **Attachment(s)**

1) Notice of References Cited (PTO-892)  
 2) Notice of Draftsperson's Patent Drawing Review (PTO-948)  
 3) Information Disclosure Statement(s) (PTO/S/65/08)  
 Paper No(s)/Mail Date 1/21/08

4) Interview Summary (PTO-413)  
 Paper No(s)/Mail Date. \_\_\_\_\_

5) Notice of Informal Patent Application

6) Other: \_\_\_\_\_

### **DETAILED ACTION**

1. This action is responsive to the Applicant's response filed 1/23/08.

As indicated in Applicant's response, claims 1, 4-6, 11, 14-16, 20 have been amended.

Claims 1-20 are pending in the office action.

#### **Double Patenting**

2. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3. Claims 5, 6, 15, 16 are provisionally rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claim 1 of copending Application No. 10,777,743 (hereinafter '743). Although the conflicting claims are not identical, they are not patentably distinct from each other because of the following observations.

Following are but a few examples as to how the certain claims from the instant invention and from the above copending application are conflicting.

**As per instant claim 5**, '743 claim 1 also recites first call tree for first build and second call tree for second build, copying a first tree data structure; subtracting the call tree for the second build from the call tree for the first build; outputting the subtracted call tree, walking the second build tree structure over the first build tree structure to generate a subtracted tree; wherein for each node that exists in both the copied call tree for the first build and the call tree for the second build, generating a node in the subtracted tree data structure by subtracting a base value of the node in the second build tree from a base value of a corresponding node in the copied first call tree. Although '743 claim 1 does not recite exact wording (e.g. first build minimized call tree structure versus first call tree structure) as instant claim 5, the difference between the respective teaching regarding for example, second call tree and second build call tree structure would have been a obvious limitation. '743 discloses traversal with specific intent to computer difference among nodes visited; but does not recite inserting a pass field in each node of the copied first tree structure and initializing such node; but based on the walking algorithm as entailed from '743, well-known practices for initializing of a node every time a node (e.g. with a flag or indicative value) is visited (in order to compute a value during this traversal as in '743) would have been a obvious implementation in '743 by which algorithmic iteration step would be prevented from visiting a node more than needed based on such indicator than varies with the iteration.

**As per instant claim 6**, '743 claim 1 also recites first call tree for first build and second call tree for second build, copying a first tree data structure; subtracting the call tree for the second build from the call tree for the first build; outputting the subtracted call tree, wherein for each node that exists only in the second build call tree structure, generating a node in the

subtracted tree data structure having a negative value corresponding to a base value of the node that exists only in the second build tree structure, including the rationale of obviousness regarding insertion and initializing of pass field limitations. Although '743 claim 1 does not recite exact wording as instant claim 6, the difference between the respective teaching regarding first build copied call tree and second build second tree structure would have been a obvious limitation.

**As per instant claim 15**, this is computer medium version of instant claim 5, reciting the same limitations therein, hence instant claim 15 would have been a obvious variation of '743 claim 1, by virtue of the above analysis.

**As per instant claim 16**, this is computer medium version of instant claim 6, reciting the same limitations therein, hence instant claim 16 would have been a obvious variation of '743 claim 1, by virtue of the above analysis.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

*Claim Rejections - 35 USC § 112*

4. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

5. Claims 1-20 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant

art that the inventor(s), at the time the application was filed, had possession of the claimed invention. As recited in claims 1, 11, 20 the scenario of having a first call tree then a copied call tree structure and using this copied version to generate a subtracted call tree is deemed a lack of teaching or insufficient support from the Disclosure.

Claim 1 recites ‘obtaining a first call tree’ then ‘copying the first call tree to form a copied call tree data structure’ then ‘generate a subtracted call tree structure’, leading thereby to the understanding that a copied first call tree is separate entity from the first call tree, that a subtracted tree is a distinct tree resulting from the subtracting process using both the second tree and the above copy (of the first tree); that is, at least 3 trees (not counting the second call tree) are necessarily part of the claimed invention: a first call tree, a copy thereof, a subtracted tree. However, based on the corresponding support from the Specification, what is interpreted as a concurrent existence of (i) first call tree, (ii) ‘copied version of a first call tree’ and (iii) ‘subtracted call tree’ is deemed not credible a teaching as to enable one of ordinary skill in the art to make use of the above teaching, or to acknowledge that the inventor does possess all the entities (i) (ii) and (iii) based on the respective order of appearance thereof in the claim.

According to the Specifications a process starts with creating a copy of a first call tree as target build A (Specifications: *generating a copy of the call tree for the first build A* - bottom half, pg. 5 to first half pg. 6; bottom pg. 34), then based on the call tree of the build B (second call tree structure), walking to compare Build A tree to Build B tree is performed. Hence, the generating of build A tree would be interpreted as a first call tree being obtained. Absent any slightest detail about how this build A tree would be used to create a duplicate tree, this description does not sufficiently teach one of ordinary skill in the art that a copying process is

performed so that all the content of this build A tree thus created is duplicated to generate another build A 'copied call tree'. Moreover, any node whose value is being set based on comparing between respective nodes in the build A tree and in build B tree happens to belong to the build A tree (first call tree), during the walking process (*value for this node is set* – bottom pg. 34; Fig. 9B); and, as disclosed, the resulting tree wherein all nodes have been set to a value would be the result of, this very build A tree. That is, there no mention of creation of a copy for build A tree, nor is there creation of a separate third tree structure (emphasis added) being actually disclosed when this node is set to a base value -- or a node is created exactly at very location where the 'copied tree' (i.e. the first call tree) is being compared (e.g. Specifications: bottom pg. 34, top pg. 35). As disclosed in the walking down wherein only two call tree structures, a Build A tree and a Build B tree, are involved (with the former being modified to yield a subtracted end resulting tree) the entities as claimed -- (i), (ii), and (iii) – are deemed not in possession by the Inventor at the time the Invention was made. Thus, the *copied tree* recited as (ii) and the obtaining of (i) would be treated as one tree from generating said first call tree, and the generating of (iii) would be treated as mere modifying of an existing tree (e.g. the first call tree) based on the comparison process between the second tree and the first tree.

Claims 11, and 20 exhibit language that entails existence of tree structures (i) and (ii) along with the subtracted tree structure (iii); hence would also be rejected for lack of enabling description based on the above analysis. The entities (ii) and (iii) would be treated as slight language variant representing to the same first call tree (i) being initially created or obtained.

Claims 2-10, 12-19 are also rejected for failing to remedy to the above lack of definite teaching. Correction is required.

6. Claims 4 and 14 are rejected for failing to comply with the written description requirement, in terms of containing subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

As recited, the *copying of first tree structure to form a copied call tree structure* **includes** the steps of (claims 4, 14) inserting a pass in said copied tree, initializing the pass for each node therein, and walking the second tree over the copied tree structure. As described in the Specifications, the process of initializing pass field and walking the Build A tree and Build B tree does not necessitate creation of a copy of build A tree; nor is any copying process (to form a copied call tree structure – as recited) described (Specifications: bottom half, pg. 5 to first half pg. 6; bottom half pg. 34) with reasonable details in order for one to construe that the steps of inserting, initializing and walking are steps integral to this very copying as claimed. In light of the lack of enabling support, the above inserting, initializing and walking would be treated as being part of the walking process; rendering what is phrased as 'copying of first tree structure to form a copied call tree structure ... includes' largely without merits or patentable weight.

#### *Claim Rejections - 35 USC § 103*

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made

8. Claims 1-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Levine et al, USPN: 6,349, 406 (hereinafter Levine) in view of Reissman et al, USPubN: 2005/00171818 (hereinafter Reissman).

**As per claim 1,** Levine discloses a method, in a data processing system, for identifying differences between the execution of a first computer program and a second computer program, comprising:

obtaining a first call tree data structure corresponding to first trace data of an execution of the first computer program (e.g. first pass - Fig. 21; *call stack tree* representation – col. 23, lines 1-8; Fig. 22);

obtaining a second call tree data structure corresponding to second trace data of an execution of the second computer program (e.g. second pass – Fig. 21 – Note: walking a stack tree representing events with entry and exit of functions **called** based on a trace record reads on a call tree structure – see col. 10, lines 15-22; col. 12, lines 15-47 – and a walking based on a different trace reads on a second call tree);

*copying the first call tree data structure to form a copied call tree data structure*  
(NOTE: the generating of a first structure tree by Levine to enable to comparison step to start would read on having created a copy of a tree to consider for walking against the second tree structure --see **interpretation as set forth in the USC § 112, 1<sup>st</sup> para Rejection**, where first tree, copy of first tree and the resulting tree – subtracted tree – are subsumed in one tree created as one initial first tree being used for the walking process)

subtracting the second call tree data structure from the first call tree data structure to generate a subtracted call tree data structure (delta event – Fig. 20A; if there are more traces –

col.18, lines 7-25 – Note: traces to process reads on first trace and second trace with corresponding call tree each ); and

outputting the subtracted call tree data structure, wherein the subtracted call tree data structure identifies differences ( step 2312 – Fig. 23A; col. 22, line 25-55 – Note: delta time node associated to node of stack tree reads on outputting delta represented by stack tree structure, hence created a *subtracted stack tree* –see step 2314, Fig. 23A) between the execution of the first computer program and the execution of the second computer program.

Levine does not explicitly disclose that the first computer program is a first build thereof and that the second computer program is second build thereof. Levine mentions about using differences that may happen for different platforms, e.g. as Java APIs -- so that trace code has to be used to effect profiling and reduction of code (col. 7, line 61 to col. 8, line 17; col. 9, line 30-48) as endeavored by developers for enhancing code for a given hardware platforms ( see BACKGROUND: col. 2, lines 50-58), hence has suggested code improvement made to accommodate software build for platforms. Reissman also discloses a tool to test APIs based on analysis of software builds via a scanning tool whereby dependencies tree is supporting recording of differences between method calls including execution state of these calls, based on such record or dictionary structure so to yield differences (see para 0041, pg. 4; Fig. 3). It would have been obvious for one skill in the art at the time the invention was made to implement the trace analysis and difference structure generating approach by Levine, so that the computer program being traced and converted into tree of events for profile are computer intended for builds as taught by Reissman, because executables intended for different environments (see Reissman) via improved builds can be addressed by profiling and analysis based on events as set

forth by either Levine and Reissman, whereby based on such differences between trace profiling data, one build as well as testing techniques therefor can be readjusted for improvement over the previous build (see Reissmann, BACKGROUND).

**As per claim 2,** Levine discloses inputting the first trace data and the second trace data to an arcflow tool (Arcflow output – Fig .17-18 ), wherein the arcflow tool generates the first call tree data structure and the second call tree data structure based on the first trace data and the second trace data.

**As per claim 3,** Levine discloses wherein the first call tree data structure and the second call tree data structure are xtree data structures (Fig. 17-18 – Note: tree involving stack data and state reads on eXecution type of tree, i.e. Xtree).

**As per claim 4,** Levine discloses for the step of forming a copied call tree structure (Note: copying step treated as process of incrementally forming the resulting subtracted tree based on the first call tree – refer to USC 112, first paragraph Rejection):

inserting a pass field (e.g. step 1352, step 1358 -Fig 13) in each node of the copied call tree data structure (copied first tree treated as first tree initialized) ;

initializing the pass field (Fig. 13-14; *Base Time* – Fig. 23B) in each node of the copied call tree data structure; and walking the second call tree data structure over the copied call tree data structure to generate the subtracted call tree data structure (e.g. delta event time - Fig. 20A-B; Fig. 21).

**As per claim 5,** Levine discloses wherein walking the second call tree data structure over the first call tree data structure includes: for each node that exists in both the copied call tree data structure and the second call tree data structure, generating a node in the subtracted call tree

data structure by subtracting a base value (e.g. Fig. 20B; *minus*, step 2312 – Fig. 23A; Base Time - Fig. 23B – Note: subtracting reads on based on values of *both* nodes in respective tree) of the node in the second call tree data structure from a base value of a corresponding node in the first call tree data structure.

**As per claim 6**, Levine discloses wherein walking the second call tree data structure over the first call tree data structure includes: for each node that exists only in the second call tree data structure, creating a node in the subtracted call tree data structure having a negative base value (Note: base value in either call tree – e.g. only in second tree -- reads on delta computing as a result of either adding to a base value; i.e. subtracting a negative value yield a added value – see CUM, Fig. 16 -- OR subtracting from a base value with a greater from the either the first or second tree value - see step 2312 – Fig. 23A; see Fig. 23B, 20B) corresponding to a base value of the node (e.g. that exists only in the second call tree data structure).

**As per claims 7-8**, Levine discloses setting a value of a pass field of the node in the subtracted call tree data structure to a value indicating that both the first call tree data structure and the second call tree data structure contributed to the base value (*delta* – Fig. 20A; 2364 *adjusted base time for routine C*, pg. 23B); setting a value of a pass field of the node in the subtracted call tree data structure to a value indicating that either the first call tree data structure or the second call tree data structure contributed to the base value (sum child self base – Fig. 17; CUM, Fig. 16).

**As per claim 9**, Levine discloses, wherein nodes in the first call tree data structure and nodes in the second call tree data structure whose paths have not changed between builds are not present in the subtracted call tree data structure (step 2312 – Fig. 23A; col. 22, line 25-55 – Note:

delta time being outputted by stack tree structure, hence created a *subtracted stack tree* – see step 2314, Fig. 23A reads on not outputting where there is no delta).

**As per claim 10,** Levine discloses identifying improvements (e.g. col. 2, lines 37-59; pruning – col. 19, lines 7-30) but does not disclose identifying regressions from the first build to the second build of the computer program based on values associated with nodes in the subtracted call tree data structure. But based on Reissman's same endeavor for tracking dependencies and changes between two builds for improving of code, and implementing test coverage (see Fig. 2, 16-17; para 0040, pg. 4; Fig. 13) the regression test for discovery of bugs between two builds is strongly implied. The rationale in claim 1 using Reissman's approach to improve one build over a previous one via test coverage or regression analysis is incorporated herein for the same reasons based on Levine's endeavor and the applicability of code comparison by Reissman, as set forth above.

**As per claim 11,** Levine discloses a computer program product in a computer readable medium for identifying differences between the execution of a first computer program and a second computer program, comprising:

first instructions for obtaining a first call tree data structure corresponding to first trace data of an execution of the first computer program;

second instructions for obtaining a second call tree data structure corresponding to second trace data of an execution of the second computer program;

third instructions for subtracting the second call tree data structure from the first call tree data structure to generate a subtracted call tree data structure; and

fourth instructions for outputting the subtracted call tree data structure, wherein the subtracted call tree data structure identifies differences between the execution of the first computer program and the execution of the second computer program;

fifth instructions for copying the first call tree data structure to form a copied call tree data structure;

all of which step actions having been addressed in claim 1.

Levine does not explicitly disclose that the first computer program is a first build thereof and that second computer program is a second build thereof. But this limitation has been addressed in claim 1.

**As per claims 12-19**, refer to claims 2-8, 10 respectively.

**As per claim 20**, Levine discloses an apparatus for identifying differences between the execution of a first computer program and a second computer program, comprising means for:

obtaining a first call tree data structure corresponding to first trace data of an execution of the first computer program;

obtaining a second call tree data structure corresponding to second trace data of an execution of the second computer program;

copying the first call tree data structure to form a copied call tree data structure;

subtracting the second call tree data structure from the first call tree data structure to generate a subtracted call tree data structure; and

outputting the subtracted call tree data structure, wherein the subtracted call tree data structure identifies differences between the execution of the first computer program and the execution of the second computer program;

all of which step actions having been addressed in claim 1.

Levine does not explicitly disclose that the first computer program is a first build thereof and that second computer program is a second build thereof. But this limitation has been addressed in claim 1.

***Response to Arguments***

9. Applicant's arguments filed 1/23/08 have been fully considered but they are not persuasive. Following are the Examiner's observation in regard thereto.

**Double Patenting Rejection:**

(A) Applicants have submitted that not until allowable subject matter is identified, would a need for providing a terminal Disclaimer be justified. The rejection has now been adjusted to address the change of claims in both co-pending Applications. The representative in a course of a brief telephonic contact, 4/3/08 has been informed that the process of approving a Terminal Disclaimer (by the Office) might take some time, and this procedural delay might be prohibitive for the Office to resolve any pending issues left for the prosecution to reach a allowable condition, should a occasion arises where a possibility (e.g. Examiner's Amendments) to expediently lead to that condition would present itself. A timely submitted Disclaimer thus contemplated or deemed necessary, would help expedite such process to reach such agreement.

**35 USC § 103 Rejection:**

(B) Applicants have submitted that burden to establish *prima facie* is on the Examiner for obviousness legal conclusion; and regarding which Levine does not teach or suggest 'copying the first call tree data structure to form a copied call tree data structure' (Appl. Rmrks pg. 11).

The added limitation has created a new ground of rejection herein as necessitated thereby, and this new grounds will render any argument to this added limitation moot.

(C) Applicants have submitted that Levine does not teach or suggest 'subtracting the second tree ... from the copied call tree ... to generate a subtracted tree ...' as amended; nor does Levine teach second tree structure (Appl. Rmrks, pg. 12, top). The Rejection has shown comparison between stack tree node and values associated thereto, in order to yield a modified tree structure. Notwithstanding the improprieties as set forth in the USC 112, Rejection (i.e. *copied call tree structure* being given only weight that it is actually the first call tree), the argument concerning Levine not teaching a second tree is not convincing. The rejection has pointed to walking a instance of stack in which called events are arranged as tree nodes (e.g. Levine: col. 12, lines 15-47), based on record of traces; such that each instance of stack being walked based on a trace record or profiling thereof reads on either a first call tree structure, or a second call tree structure; e.g. because the latter is based on a second record, and the former is based on a first pass and first trace. What is referred to as 'second call tree structure' without further details in the claim is deemed not sufficiently particular in order to preclude Levine's second stack tree walking using another trace as identified in the Office action from being analogized thereto, when the first pass has been analogized to being a first tree call structure. Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the reference.

(D) Applicants have submitted that performance statistics based on trace events by Levine and computations based on trace records as such cannot disclose or suggest call tree structure

being second call tree and copied call tree in the recited 'subtracting the second call tree ... from the copied call tree ...' as amended (Appl. Rmrks pg 13, top). The Rejection has interpreted the subtracting in terms of comparing in order to propagate **change to values** in the respective nodes being walked through within the stack tree pass for identifying delta information; and when the claim does not provide further specific about how subtracting is done, the comparing and modifying of values by Levine is deemed fulfilling the subtracting limitation. Since *subtracting* is recited in a context rather devoid of quantities whatsoever so to effectuate a minimal form of computation, the limitation amounts to a broad interpretation; thus, Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the reference. As for the 'second tree' and 'copied tree' limitations, the argument concerning these, would have to be referred back the sections B and C.

(E) Applicants have submitted that Reissman does not cure to the deficiency of any reference fulfilling 'copying the first call tree...', especially when Reissman is teaching *ictionaries* and comparing these ( Appl. Rmrks pg. 13, middle, pg. 14). The argument is not commensurate with Office Action specifics for set forth the grounds of rejection. That is, the specific features being addressed relate to *build* version of codes, and in terms of which, Reissman's teaching has been applied to render obviousness. Reissman is not purported to teach 'copying' as claimed, because this *copying* is a newly added limitation. Besides, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642

F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

(F) Applicants have submitted that the advantage proffered by the Examiner fails to connect or comport with an advantage according to a legal conclusion of obviousness (Appl. Rmrks pg. 14-15) but amounts to a mere assumption without a underpinning to achieve this *legal conclusion* as required by KSR Int'l. The above argument does not establish how the two references cannot be combined in terms of analogous arts; nor does it establish whether there is insufficient suggestion in either reference; nor does it establish an impermissible use of hindsight; nor does it establish the negative result because of the assumption made when combining the reference. More importantly, it is noted that the argument, while raising doubt about the advantage from the obviousness rationale, does not dissect each reference in line with the very flow specified in the rationale set forth by the Office Action -- specifics flowing from what has been identified not disclosed by Levine (emphasis added), any suggestion there is, the existing facts, act and motivation to combine, as well as the benefits resulting therefrom (refer to Office Action); nor does the argument, based on ALL of the above (e.g. on the basis of what is not taught in Levine), derive a proper reason showing how the so 'assumed' advantage would be improper. The argument is clearly not providing a sound *prima facie* case for efficiently rebut the conclusion made in the § 103 rejection; and is deemed out of line with respect to compliant and effective format to overcome this type of rejection.

In all, the claims will stand rejected as set forth in the Office Action.

***Conclusion***

10. **THIS OFFICE ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (571) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lewis Bullock can be reached on (571)272-3759.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 ( for non-official correspondence - please consult Examiner before using) or 571-273-8300 ( for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Tuan A Vu/

Primary Examiner, Art Unit 2193

April 04, 2008